A LOSSLESS IMAGE CODING TECHNIQUE EXPLOITING SPECTRAL CORRELATION ON THE RGB SPACE

Stefano Andriani, Giancarlo Calvagno, Gian Antonio Mian

Department of Information Engineering, University of Padova
Via Gradenigo, 6/B - 35131 Padova - Italy
e-mail: {mutley,calvagno,mian}@dei.unipd.it

ABSTRACT

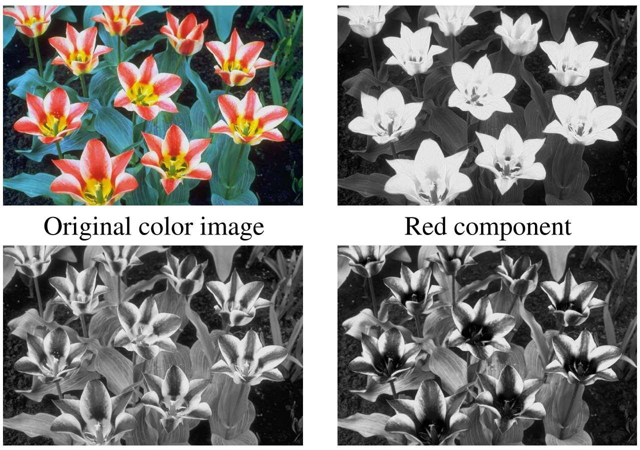
In this paper we exploit spectral correlation of color images in the RGB space for lossless image coding. We propose a simple way to use this correlation in a predictive model by correcting the prediction in one band by means of the prediction error in the adjacent band.

The experimental results show that this method gives a significant performance improvement over all the tested predictors.

1. INTRODUCTION

Lossless compression of images became in the last years an important option for several applications like scientific and medical images.

While the state-of-the-art for lossy compression has reached a quite mature stage, lossless compression is an hot topic that is now being considered for color images and video coding [1]. The majority of lossless compression algorithms are developed for grey-scale images and they only use the *spatial correlation*. So when these algorithms are used to code color images they do not exploit the *correlation between adjacent spectral bands*. This correlation is very useful when working on the RGB space because the three bands are strongly correlated (see Figure 1).



Original color image — Red component — Green component — Blue component

Figure 1: Spectral correlation in RGB space of Tulips image.

This paper explores the possibility to exploit the RGB space spectral correlation by performing a prediction correction using the prediction error of adjacent bands. The proposed technique gives significant improvements in performance with a low computational cost.

The paper is structured as follows: in Section 2 we review a method presented by Memon and Sayood in [2] and we propose some modifications to improve its performances. Section 3 presents the GLICBAWLS algorithm [3] and some ways to reduce its complexity [4]. In Section 4, the prediction correction by spectral correlation is presented in detail. Finally, conclusions are drawn in Section 5.

2. LOSSLESS COMPRESSION BY MEMON AND SAYOOD ALGORITHM

Memon and Sayood [2] use spectral correlation to select the best fixed predictor among the set of JPEG lossless mode [5] predictors in a color band. Then, they use it to predict the adjacent band current pixel.

The results of fixed predictors are reported in Table 1 for $3 \times 8 = 24$ bit/pixel RGB images. Predictor 6 is the best of the set (on the average) when coding is performed using a *fixed* predictor for coding the three R, G and B components. Memon-Sayood algorithm (9^{th} column of Table 1) improved by 1.4 bit per pixel the results obtained by the best JPEG lossless predictor. In the same table (10^{th} column) we have also reported the results obtained including the MED *adaptive* predictor used in JPEG-LS [6]

$$\hat{p}_{x,y} = \begin{cases} \min(a,b) & \text{if } c \geq \max(a,b) \\ \max(a,b) & \text{if } c \leq \min(a,b) \\ a+b-c & \text{otherwise} \end{cases}$$

where $a = p_{x-1,y}$, $b = p_{x,y-1}$ and $c = p_{x-1,y-1}$. The results show an improvement of 0.14 bit/pixel over the original version of the algorithm.

3. GLICBAWLS ALGORITHM

GLICBAWLS is an acronym for Grey Level Image Compression By Adaptive Weighted Least Squares. It is a prediction based coder introduced by Meyer and Tischer in [3]. At the beginning it was developed to work on grey-scale images and, after, the authors proposed a version which was able to code color images. This algorithm achieves the best compression among all the available methods but at a high computational cost.

GLICBAWLS constructs the optimal linear causal predictor of the current pixel using the twelve neighbor pixels with Manhattan distance ≤ 3 (in the following referred to as P12). New weights are calculated for each pixel of the image, taking into account all the pixels already coded.

Image	JPEG lossless mode predictor number							MED	Memon Sayood	Memon-Sayood with Med in the set of predictors
	1	2	3	4	5	6	7			
Lena	15.64	14.61	16.11	15.31	14.91	14.52	14.40	14.41	14.21	14.12
Peppers	13.80	14.18	15.37	13.23	13.06	13.10	13.19	12.60	12.03	11.83
Monarch	13.68	13.77	14.62	13.03	12.82	12.87	12.74	12.46	11.33	11.23
Sail	16.99	17.88	18.80	17.13	16.55	16.84	16.70	16.29	14.30	14.16
Tulips	15.33	14.83	16.39	13.86	13.96	13.71	14.13	13.48	12.48	12.33
Average	15.08	15.05	16.26	14.51	14.26	14.21	14.23	13.85	12.87	12.73

Table 1: Prediction error entropy on a set of test images (in Bit per Pixel).

Given the autocorrelation's matrix $\mathbf{A}_i$ of pixel i and the vector $\mathbf{b}_i$ that contains R, G or B values we compute the matrix

$$\mathbf{A_C} = \sum_{i=1}^{N} 0.8^{|x_C - x_i| + |y_C - y_i|} \mathbf{A_i} \quad (1)$$

and the vector

$$\mathbf{b_C} = \sum_{i=1}^{N} 0.8^{|x_C - x_i| + |y_C - y_i|} \mathbf{b_i} \quad (2)$$

where the factor $0.8^{|x_C - x_i| + |y_C - y_i|}$ is used to give less weight to the pixels with higher distance from the current position (pixel). In (1) and (2), N is the number of previously coded pixels. The prediction coefficients are calculated solving the linear system

$$\mathbf{A_C w} = \mathbf{b_C}. \quad (3)$$

From this equation we can calculate the predicted value for the current pixel

$$\hat{p}_{x,y} = \sum_{k=1}^{12} w_k n_k$$

with n_k the R, G or B value of the causal pixel k with Manhattan distance ≤ 3 from the current one.

The pixel value is then coded through an arithmetic binary coder in bit's plane mode. It starts from the MSB down to the LSB and the zero's probability is computed modelling the prediction error by a modified Student distribution centered on the predicted pixel value [3].

Linear prediction is computationally demanding because for each pixel the algorithm solves one linear system of twelve equations in twelve unknowns for each R, G and B component. The complexity is $\mathscr{O}(n^3/6)$ using the Chowlesky algorithm to solve system (3).

One idea to reduce complexity is to exploit spectral correlation and to reuse the red band coefficients in the adjacent bands (green and blue). In this way we use the optimal predictor in the first band and sub-optimal predictors in the other bands [4]. So, the GLICBAWLS performance is slightly reduced (on the average about 0.1 bit/pixel: see first two columns of Table 2) but the algorithm becomes about two times faster.

Another way to reduce the GLICBAWLS complexity is to calculate the optimal linear predictor on the six pixels which have Manhattan distance ≤ 2 with respect to the current pixel (referred to as P6). In this way the algorithm

Image	12 pixels predictors		6 pixels predicotors	
		Reuse		Reuse
Lena	12.75	12.81	12.87	12.93
Peppers	11.19	11.34	11.28	11.43
Monarch	10.86	10.92	11.01	11.07
Sail	14.67	14.70	15.09	15.12
Tulips	11.58	11.91	11.82	12.09
Average	12.21	12.33	12.42	12.54

Table 2: Modified GLICBAWLS algorithms.

"only" solves three linear systems of six equations in six unknowns. This version has a computational complexity five times smaller than the original. Also in this version we can reuse the optimal coefficients of red band in the other bands.

The results (in bit/pixel) of the algorithms are reported in Table 2. We can observe that we lose a maximum of 0.33 bit/pixel in going from the first to the fourth column but, correspondingly, we reduce ten times the algorithm complexity. Now it is very important to underline that, while Table 1 reports the first order entropy of prediction error, Table 2 gives the actual bit per pixel value obtained.

4. PREDICTION CORRECTION BY SPECTRAL CORRELATION

Another simple and effective way to exploit the spectral correlation is to correct the predicted pixel value in one band using the prediction error of the adjacent component.

More precisely the prediction can be corrected (see Figure 2) using the following procedure.

- The algorithm predicts the red band current pixel $\hat{x}_R$ and the prediction error

  $$e_R = x_R - \hat{x}_R$$

  is coded as described in Section 3.
- The adjacent green component current pixel is predicted as $\hat{x}_G$, and the prediction error

  $$e_G = x_G - \hat{x}_G$$

  is computed. The predicted value $\hat{x}_G$ is then corrected by subtracting the red band prediction error e_R. In this way the corrected predicted value becomes

  $$\tilde{x}_G = \hat{x}_G - e_R$$

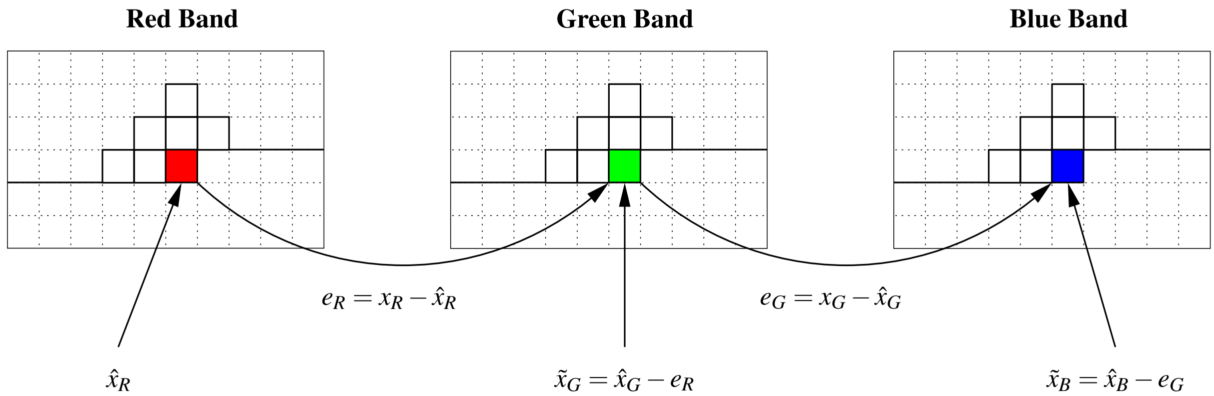


Figure 2: Prediction correction by prediction error of adjacent band.

Image	RGB independently coded			
	12 pixels pred.		6 pixels pred.	
		With corr.		With corr.
Lena	12.75	12.57	12.87	12.63
Peppers	11.19	9.93	11.28	9.96
Monarch	10.86	9.24	11.01	9.21
Sail	14.67	11.34	15.09	11.40
Tulips	11.58	10.14	11.82	10.17
Average	12.21	10.64	12.42	10.87

Table 3: GLICBAWLS algorithm applied independently on the three spectral bands with or without prediction correction.

and the prediction error

$$\tilde{e}_G = x_G - \tilde{x}_G = x_G - \hat{x}_G + e_R.$$

The current pixel x_G is coded centering the modified Student distribution on $\tilde{x}_G$.

- The algorithm predicts the current pixel on the blue band and it applies a correction similar to that used for the green band. The green band prediction error e_G (made before the correction) is used to correct the blue band (Figure 2).

The correction introduced has a very low computational cost and it can be applied to *any* predictor. It is effective since the correction sharpens the estimated prediction error distribution hence reducing the underlying entropy as shown in Figure 3 and in Figure 4.

The proposed prediction correction method improves the GLICBAWLS performance of 1.55 bit/pixel when the R, G and B components are independently coded for both P12 and P6, as shown in Table 3.

Table 5 shows the results obtained when the prediction correction is applied on the GLICBAWLS version that reuses the optimal coefficients of the red band on the other bands. In this case a greater gain of about 1.9 bit/pixel is obtained, i.e., the performance increases when the prediction correction is applied to the reduced complexity algorithms. Comparing 2^{th} and 4^{th} columns of Tables 3 and 5 it results that there is a

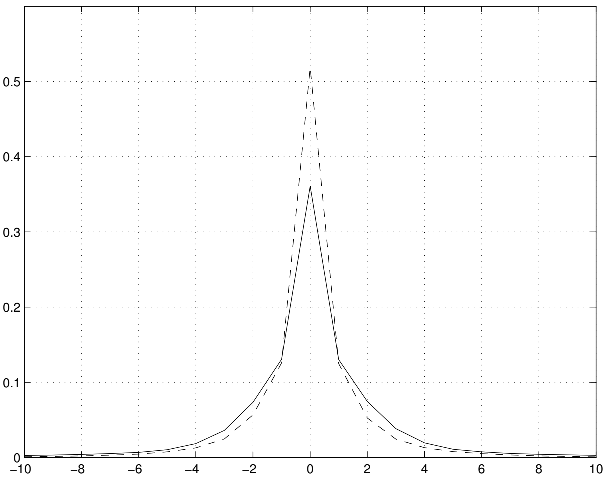


Figure 3: Prediction error distribution without (solid line) and with (dashed line) prediction correction of the green band of Monarch image.

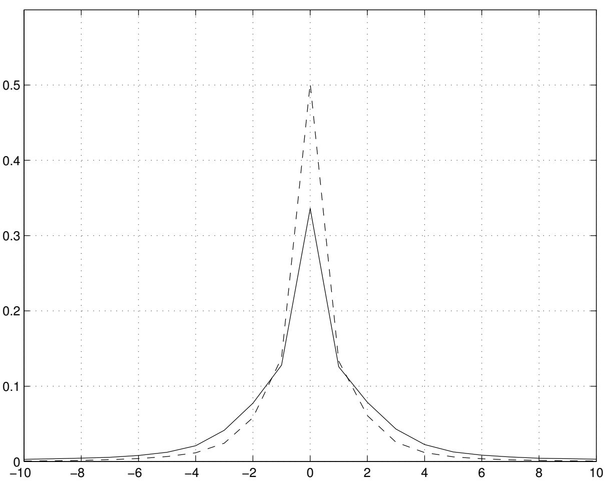


Figure 4: Prediction error distribution without (solid line) and with (dashed line) prediction correction of the blue band of Monarch image.

Image	JPEG lossless mode predictor number							MED	Memon Sayood	Memon-Sayood with Med in the set of predictors
	1	2	3	4	5	6	7			
Lena	14.85	14.06	15.25	14.97	14.40	14.13	13.75	14.04	14.08	14.00
Peppers	12.17	13.19	14.26	11.18	11.27	11.66	11.87	11.15	11.56	11.34
Monarch	11.84	11.99	13.06	10.21	10.55	10.64	10.88	10.37	10.58	10.48
Sail	12.91	13.33	14.38	11.79	11.83	11.97	12.11	11.87	12.22	12.08
Tulips	13.23	12.84	14.42	11.02	11.61	11.39	12.02	11.33	11.75	11.60
Average	13.00	13.08	14.27	11.83	11.88	11.96	12.13	11.75	12.04	11.90

Table 4: Prediction error entropy in a set of static image (in Bit per Pixel) using prediction correction.

Image	Reuse coefficients of red band on others bands			
	12 pixels pred.		6 pixels pred.	
		With corr.		With corr.
Lena	12.81	12.51	12.93	12.63
Peppers	11.34	9.99	11.43	10.05
Monarch	10.92	9.12	11.07	9.21
Sail	14.70	10.83	15.12	10.98
Tulips	11.91	10.20	12.09	10.26
Average	12.33	10.53	12.54	10.63

Table 5: GLICBAWLS algorithm with reuse of red prediction coefficients on the other bands without and with prediction correction.

0.11 bit/pixel improvement in the P12 case and a greater 0.24 bit/pixel improvement in the P6 case. The reason for this improvement is that when reusing coefficients of red band we introduce additional correlation between prediction errors on different bands. So, the proposed prediction correction exploits this additional correlation to cancel the performance loss reported in Table 2 and explained in Section 3.

To show the effectiveness of the prediction correction method independently of the predictor used, Table 6 reports the results obtained applying it to the LOCO-I algorithm [6]. The prediction correction improves the LOCO-I performance by 1.66 bit/pixel.

Finally we apply the method to correct the prediction of the JPEG lossless predictors introduced in Section 2. The results are reported in Table 4 where we can notice an average improvement of about 2 bits/pixel with respect to the results of Table 1. In this case the Memon-Sayood algorithm is not the best one because the simpler *fixed* MED predictor with prediction error correction did better on the average. Possibly, in this last case, the prediction errors are more correlated so the spectral correction introduced works better.

5. CONCLUSION

In this work we presented a new method to exploit the spectral correlation in lossless prediction based color image coders. The procedure gives good performance improvement independently from the used predictor and its complexity is very low.

Image	Without correction	With correction
Lena	13.53	13.53
Peppers	11.76	10.56
Monarch	11.31	9.84
Sail	15.51	11.67
Tulips	12.54	10.74
Average	12.93	11.27

Table 6: LOCO-I algorithm with and without correction of prediction by prediction error of adjacent band.

This method should also give good performance on multi-spectral images where the adjacent bands are closer than the RGB bands, so that they are more correlated and therefore the prediction error of adjacent bands is more correlated too.

REFERENCES

[1] D. Brunello,G. Calvagno, G.A.Mian, R. Rinaldo, “Lossless video coding using temporal information”, *IEEE Transaction on Image Processing*, Vol. 12, No. 2, pp. 132-139, February 2003.

[2] N. D. Memon, K. Sayood, “Lossless Compression of Video Sequences”, *IEEE Transaction on Communications*, Vol. 44(10):1340-1345, October 1996.

[3] B. Meyer, P. Tischer, “Glicbawls - Gray Level Image Compression By Adaptive Weighted Least Squares”, *Proc. of Data Compression Conference*, Piscataway, NJ, USA, March 2001.

[4] S. Andriani, “*Codifica reversibile di sequenze di immagini a colori*”, Laurea degree thesis, Dept. of Information Engineering, University of Padova, Italy, April 2003.

[5] G. K. Wallace, “The JPEG still picture compression standard”, *IEEE Trans. Consumer Electronics*, vol. 38, pp. 28-34, February 1993.

[6] M. J. Weinberger, G. Seroussi, G. Sapiro, “The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS”, *IEEE Trans. on Image Processing*, Vol. 9, No. 8, August 2000.